

- [11] G. Ungerboeck, "Channel coding with multilevel/phase signals," *IEEE Trans. Inform. Theory*, vol. IT-28, pp. 55-67, Jan. 1982.
- [12] —, "Trellis-coded modulation with redundant signal sets, Parts I and II," *IEEE Commun. Mag.*, vol. 25, no. 2, pp. 5-21, Feb. 1987.
- [13] L.-F. Wei, "Trellis-coded modulation with multidimensional constellations," *IEEE Trans. Inform. Theory*, vol. IT-33, pp. 483-501, 1987.

## A Generalized Convergence Theorem for Neural Networks

JEHOSHUA BRUCK AND JOSEPH W. GOODMAN, FELLOW, IEEE

**Abstract**—A neural network model is presented in which each neuron performs a threshold logic function. An important property of the model is that it always converges to a stable state when operating in a serial mode and to a cycle of length at most 2 when operating in a fully parallel mode. This property is the basis for the potential applications of the model, such as associative memory devices and combinatorial optimization. The two known convergence theorems (for serial and fully parallel modes of operation) are reviewed, and a general convergence theorem is presented which unifies the two known cases. Some new applications of the model for combinatorial optimization are also presented, in particular, new relations between the neural network model and the problem of finding a minimum cut in a graph.

### I. INTRODUCTION

The neural network model is a discrete-time system that can be represented by a weighted and undirected graph. A weight is attached to each edge of the graph and a threshold value attached to each node (neuron) of the graph. The order of the network is the number of nodes in the corresponding graph. Let  $N$  be a neural network of order  $n$ ; then  $N$  is uniquely defined by  $(W, T)$  where

- $W$  is an  $n \times n$  symmetric matrix, where  $W_{ij}$  is equal to the weight attached to edge  $(i, j)$ ;
- $T$  is a vector of dimension  $n$ , where  $T_i$  denotes the threshold attached to node  $i$ .

Every node (neuron) can be in one of two possible states, either 1 or  $-1$ . The state of node  $i$  at time  $t$  is denoted by  $V_i(t)$ . The state of the neural network at time  $t$  is the vector  $V(t)$ .

The next state of a node is computed by

$$V_i(t+1) = \text{sgn}(H_i(t)) = \begin{cases} 1, & \text{if } H_i(t) \geq 0; \\ -1, & \text{otherwise} \end{cases} \quad (1)$$

where

$$H_i(t) = \sum_{j=1}^n W_{ij} V_j(t) - T_i.$$

The next state of the network, i.e.,  $V(t+1)$ , is computed from the current state by performing the evaluation (1) at a set  $S$  of the nodes of the network. The modes of operation are determined by the method by which the set  $S$  is selected in each time interval. If the computation is performed at a single node in any time interval, i.e.,  $|S|=1$ , then we say that the network is operating

in a *serial* mode; if  $|S|=n$ , then we say that the network is operating in a *fully parallel* mode. All the other cases, i.e.,  $1 < |S| < n$ , will be called *parallel* modes of operation. The set  $S$  can be chosen at random or according to some deterministic rule.

A state  $V(t)$  is called *stable* if and only if  $V(t) = \text{sgn}(WV(t) - T)$ , i.e., no change occurs in the state of the network regardless of the mode of operation.

An important property of the model is that it always converges to a stable state when operating in a serial mode and to a cycle of length at most 2 when operating in a fully parallel mode [3], [5]. Section II contains a description of these convergence properties and a general convergence theorem which unifies the two known cases. New relations between the energy functions which correspond to the serial and fully parallel modes are presented as well.

The convergence properties are the basis for the application of the model in combinatorial optimization. In section III we describe the potential applications of a neural network model as a local search device for the two modes of operation, that is, serial mode and fully parallel mode. In particular, we show that an equivalence exists between finding a maximal value of the energy function and finding a minimum cut in an undirected graph, and also that a neural network model can be designed to perform a local search for a minimum cut in a directed graph.

### II. CONVERGENCE THEOREMS

An important property of the model is that it always converges, as summarized by the following theorem.

**Theorem 1:** Let  $N = (W, T)$  be a neural network, with  $W$  being a symmetric matrix; then the following hold.

1) *Hopfield* [5]: If  $N$  is operating in a serial mode and the elements of the diagonal of  $W$  are nonnegative, the network will always converge to a stable state (i.e., there are no cycles in the state space).

2) *Goles* [3]: If  $N$  is operating in a fully parallel mode, the network will always converge to a stable state or to a cycle of length 2 (i.e., the cycles in the state space are of length  $\leq 2$ ).

The main idea in the proof of the two parts of the theorem is to define a so-called *energy function* and to show that this energy function is nondecreasing when the state of the network changes. Since the energy function is bounded from above, the energy will converge to some value. Note that, originally, the energy function was defined so that it is nonincreasing [3], [5]; we changed it to be nondecreasing in accordance with some known graph problems (see, e.g., min cut in the next section).

The second step in the proof is to show that constant energy implies in the first case a stable state and in the second a cycle of length  $\leq 2$ . The energy functions defined for each part of the proof are different:

$$E_1(t) = V^T(t) W V(t) - (V(t) + V(t))^T T$$

$$E_2(t) = V^T(t) W V(t-1) - (V(t) + V(t-1))^T T \quad (2)$$

where  $E_1(t)$  and  $E_2(t)$  denote the energy functions related to the first and second part of the proof.

An interesting question is whether two different energy functions are needed to prove the two parts of Theorem 1. A new result is that convergence in the fully parallel mode can be proven using the result on convergence for the serial mode of operation. For the sake of completeness, the proof for the case of a serial mode of operation follows.

**Proof of the First Part of Theorem 1:** Using the definitions in (1) and (2), let  $\Delta E = E_1(t+1) - E_1(t)$  be the difference in the energy associated with two consecutive states, and let  $\Delta V_k$  denote the difference between the next state and the current state of

Manuscript received June 1987; revised November 1987. This work was supported in part by the Rothschild Foundation and by the U.S. Air Force Office of Scientific Research. This correspondence was presented in part at the IEEE First International Conference on Neural Networks, San Diego, CA, June 1987.

The authors are with the Information Systems Laboratory, Department of Electrical Engineering, Durand Building, Stanford University, Stanford, CA 94305.

IEEE Log Number 8824061.

node  $k$  at some arbitrary time  $t$ . Clearly,

$$\Delta V_k = \begin{cases} 0, & \text{if } V_k(t) = \text{sgn}(H_k(t)) \\ -2, & \text{if } V_k(t) = 1 \text{ and } \text{sgn}(H_k(t)) = -1 \\ 2, & \text{if } V_k(t) = -1 \text{ and } \text{sgn}(H_k(t)) = 1 \end{cases} \quad (3)$$

By assumption (serial mode of operation) the computation (1) is performed only at a single node at any given time. Suppose this computation is performed at an arbitrary node  $k$ ; then the energy difference resulting from this computation is

$$\Delta E = \Delta V_k \left( \sum_{j=1}^n W_{k,j} V_j + \sum_{i=1}^n W_{i,k} V_i \right) + W_{k,k} \Delta V_k^2 - 2 \Delta V_k T_k. \quad (4)$$

From the symmetry of  $W$  and the definition of  $H_k$  it follows that

$$\Delta E = 2 \Delta V_k H_k + W_{k,k} \Delta V_k^2. \quad (5)$$

Hence since  $\Delta V_k H_k \geq 0$  and  $W_{k,k} \geq 0$ , it follows that  $\Delta E \geq 0$  for every  $k$ . Thus since  $E_1$  is bounded from above, the value of the energy will converge.

The second step in the proof is to show that convergence of the energy implies convergence to a stable state. The following two simple facts are helpful in this step:

- 1) if  $\Delta V_k = 0$ , then it follows that  $\Delta E = 0$ ;
- 2) if  $\Delta V_k \neq 0$ , then  $\Delta E = 0$  only if the change in  $V_k$  is from  $-1$  to  $1$ , with  $H_k = 0$ .

Hence once the energy in the network has converged it is clear from the foregoing facts that the network will reach a stable state after at most  $n^2$  time intervals.

The following lemma describes a general result which enables transformation of a neural network with nonnegative self-loops operating in a serial mode to an equivalent network without self-loops (part a); it also enables transformation of a neural network operating in a fully parallel mode to an equivalent network operating in a serial mode (part b). The equivalence is in the sense that it is possible to derive the state of one network given the state of the other network, provided the two networks started from the same initial state.

**Lemma 1:** Let  $N = (W, T)$  be a neural network. Let  $\hat{N} = (\hat{W}, \hat{T})$  be obtained from  $N$  as follows:  $\hat{N}$  is a bipartite graph, with

$$\hat{W} = \begin{pmatrix} 0 & W \\ W & 0 \end{pmatrix} \quad \text{and} \quad \hat{T} = \begin{pmatrix} T \\ T \end{pmatrix}.$$

We make the following claims:

- a) for any serial mode of operation in  $N$ , there exists an equivalent serial mode of operation in  $\hat{N}$  provided  $W$  has a nonnegative diagonal;
- b) there exists a serial mode of operation in  $\hat{N}$  which is equivalent to a fully parallel mode of operation in  $N$ .

**Proof:** The new network  $\hat{N}$  is a bipartite graph with  $2n$  nodes; the set of nodes of  $\hat{N}$  can be subdivided into two sets: let  $P_1$  and  $P_2$  denote the set of the first and the last  $n$  nodes, respectively. Clearly, no two nodes of  $P_1$  (or  $P_2$ ) are connected by an edge; that is, both  $P_1$  and  $P_2$  are independent sets of nodes in  $\hat{N}$  (an independent set of nodes in a graph is a set of nodes in which no two nodes are connected by an edge). Another observation is that  $P_1$  and  $P_2$  are symmetric in the sense that a node  $i \in P_1$  has an edge set similar to that of a node  $(i+n) \in P_2$ .

**Proof of a):** Let  $V_0$  be an initial state of  $N$ , and let  $(i_1, i_2, \dots)$  be the order by which the states of the nodes are evaluated in a

serial mode in  $N$ . We will show that starting from the initial state  $(V_0, V_0)$  in  $\hat{N}$  (the state of both  $P_1$  and  $P_2$  is  $V_0$ ) and using the order  $(i_1, (i_1+n), i_2, (i_2+n), \dots)$  for the evaluation of states implies the following.

1) The state of  $P_1$  will be equal to the state of  $P_2$  in  $\hat{N}$  after an arbitrary even number of evaluations.

2) The state of  $N$  at time  $k$  is equal to the state of  $P_1$  at time  $2k$ , for an arbitrary  $k$ .

The proof of 1) is by induction. Given that at some arbitrary time  $k$  the state of  $P_1$  is equal to the state of  $P_2$ , it will be shown that after performing the evaluation at node  $i$  and then at node  $(n+i)$  the states of  $P_1$  and  $P_2$  remain equal. There are two cases:

- if the state of node  $i$  does not change as a result of evaluation, then by the symmetry of  $N$  there will be no change in the state of node  $(n+i)$ ;
- if there is a change in the state of node  $i$ , then because  $\hat{W}_{i,n+i}$  is nonnegative it follows that there will be a change in the state of node  $(n+i)$  (the proof is straightforward and is not presented).

The proof of 2) follows from 1): by 1) the state of  $P_1$  is equal to the state of  $P_2$  right before the evaluation at a node of  $P_1$ . The proof is by induction: assume that the current state of  $N$  is the same as the state of  $P_1$  in  $\hat{N}$ . Then an evaluation performed at a node  $i \in P_1$  will have the same result as an evaluation performed at node  $i \in N$ .

**Proof of b):** Let us assume as in part a) that  $\hat{N}$  has the initial state  $(V_0, V_0)$ . Clearly, performing the evaluation at all nodes belonging to  $P_1$  (in parallel) and then at all nodes belonging to  $P_2$  and continuing with this alternating order is equivalent to a fully parallel mode of operation in  $N$ . The equivalence is in the sense that the state of  $N$  is equal to the state of the subset of nodes (either  $P_1$  or  $P_2$ ) of  $\hat{N}$  at which the last evaluation was performed. A key observation is that  $P_1$  and  $P_2$  are independent sets of nodes, and a parallel evaluation at an independent set of nodes is equivalent to a serial evaluation of all the nodes in the set [1]. Thus the fully parallel mode of operation in  $N$  is equivalent to a serial mode of operation in  $\hat{N}$ .

Using the transformations suggested by the foregoing lemma it is possible to explore some of the relations between convergence properties as summarized by the following theorem.

**Theorem 2:** Let  $N = (W, T)$  be a neural network. Given 1), then 2) and 3) hold:

- 1) if  $N$  is operating in a serial mode and  $W$  is a symmetric matrix with zero diagonal, then the network will always converge to a stable state;
- 2) if  $N$  is operating in a serial mode and  $W$  is a symmetric matrix with nonnegative elements on the diagonal, then the network will always converge to a stable state;
- 3) if  $N$  is operating in a fully parallel mode, then for an arbitrary symmetric matrix  $W$  the network will always converge to a stable state or a cycle of length 2; that is, the cycles in the state space are of length  $\leq 2$ .

**Proof:** The proof is based on Lemma 1.

**Part 2) is implied by Part 1):** By Lemma 1 part a every neural network with nonnegative diagonal matrix  $W$  which is operating in a serial mode can be transformed to an equivalent network to be denoted by  $\hat{N}$  which is operating in a serial mode with  $\hat{W}$  being a zero diagonal matrix.  $\hat{N}$  will converge to a stable state (by (1)); hence  $N$  will also converge to a stable state which will be equal to the state of  $P_1$ . Note that 1) is trivially implied by 2).

**Part 3) is implied by Part 1):** By Lemma 1 part b every neural network operating in a fully parallel mode can be transformed to an equivalent neural network to be denoted by  $\hat{N}$  which is operating in a serial mode and with  $\hat{W}$  being a zero-diagonal matrix.  $\hat{N}$  will converge to a stable state (by (1)). When  $\hat{N}$

reaches a stable state there are two cases:

- 1) the state of  $P_1$  is equal to the state of  $P_2$ ; in this case  $N$  will converge to a stable state which is equal to the state of  $P_1$ ;
- 2) The states of  $P_1$  and  $P_2$  are distinct; in this case  $N$  will oscillate between the two states defined by  $P_1$  and  $P_2$ , i.e.,  $N$  will converge to a cycle of length 2.

It is also interesting to investigate the relations between the two energy functions in a neural network operating in a fully parallel mode or in a serial mode. New results concerning this question are summarized in the following theorem.

**Theorem 3:** Let  $N = (W, T)$  be a neural network. Then the following hold.

- 1) For  $N$  operating in a serial mode, and for all  $t$ :

$$E_1(t-1) \leq E_2(t) \leq E_1(t).$$

- 2) For  $N$  operating in fully parallel mode, and for all  $t$ :

$$E_2(t) \geq E_1(t-1)$$

$$E_2(t) \geq E_1(t), \quad \text{when the network is in a cycle of length two}$$

$$E_2(t) \leq E_1(t), \quad \text{when the network is in a stable state.}$$

*Proof:* By the symmetry of  $W$ ,

$$\begin{aligned} E_2(t) - E_1(t-1) &= V^T(t) W V(t-1) - [V(t) + V(t-1)]^T T \\ &\quad - V^T(t-1) W V(t-1) \\ &\quad + [V(t-1) + V(t-1)]^T T \\ &= [V^T(t-1) W - T^T] [V(t) - V(t-1)]. \end{aligned} \quad (6)$$

Also,

$$\begin{aligned} E_2(t) - E_1(t) &= V^T(t) W V(t-1) - [V(t) + V(t-1)]^T T \\ &\quad - V^T(t) W V(t) + [V(t) + V(t)]^T T \\ &= [W V(t) - T]^T [V(t-1) - V(t)]. \end{aligned} \quad (7)$$

- 1) Assume without loss of generality that the evaluation at time  $t$  is performed at node 1; hence

$$V_i(t) - V_i(t-1) = 0, \quad \text{for } i \neq 1$$

$$V_1(t) - V_1(t-1) \text{ is either } 0 \text{ or } 2V_1(t).$$

By definition,

$$V_1(t) = \text{sgn} \left( \sum_{j=1}^n W_{j,1} V_j(t-1) - T_1 \right) = \text{sgn} [H_1(t-1)]. \quad (8)$$

By the previous fact the value of (6) will be either 0 or  $2H_1(t)$ ; thus it follows that

$$E_2(t) \geq E_1(t-1).$$

Using the same arguments on (7), we have

$$V_i(t-1) - V_i(t) = 0, \quad \text{for } i \neq 1$$

$$V_1(t-1) - V_1(t) \text{ is either } 0 \text{ or } -2V_1(t).$$

In this case we also have to assume that  $W$  has a nonnegative diagonal so that the following is true:

$$V_1(t) = V_1(t+1) = \text{sgn} [H_1(t)]. \quad (9)$$

By the foregoing facts it follows that (7) is either 0 or  $-2H_1(t)$ , i.e.,  $E_1(t) \geq E_2(t)$ .

- 2) By (1) for a fully parallel mode,

$$V(t) = \text{sgn} (V^T(t-1) W - T). \quad (10)$$

Since the sign of  $(V_i(t) - V_i(t-1))$  can be either 0 or  $V_i(t)$ , it follows that the difference in the two energies (6) is nonnegative,

i.e.,  $E_2(t) \geq E_1(t-1)$ . To prove the last two inequalities we will use (7). By definition,

$$V(t+1) = \text{sgn} [W V(t) - T].$$

On a cycle of length 2,  $V(t-1) = V(t+1)$ ; thus  $E_2(t) \geq E_1(t)$ . In a stable state,  $V(t+1) = V(t)$ ; thus  $E_1(t) \geq E_2(t)$ .

Some remarks regarding the foregoing analysis follow.

- 1) In a network operating in a serial mode both  $E_1$  and  $E_2$  are nondecreasing. Furthermore, a very interesting result (Theorem 3 part 1) is that the values of  $E_1$  and  $E_2$  are interleaving.

- 2) The assumption of  $W$  being a nonnegative diagonal matrix is used to derive results for a network operating in a serial mode only.

- 3) In a network operating in a fully parallel mode,  $E_2$  is nondecreasing [3]. However,  $E_1$  is not necessarily nondecreasing; it can be shown that a sufficient condition for  $E_1$  to be nondecreasing is that  $W$  is nonnegative definite over the range  $(-1, 0, 1)$ .

- 4) The proof of Theorem 3 is based on a straightforward algebraic operations. It turns out that Theorem 3 can also be proven by using Lemma 1 and the fact that the energy  $E_1$  is nondecreasing in a network operating in a serial mode (Theorem 1 part 1). We include a sketch of the alternative proof to emphasize the power of Lemma 1 in understanding the relations between the two energy functions and the two modes of operation. In the following proofs we will use the notation established in Lemma 1.

*Proof of Theorem 3 part 1:* Perform the transformation of  $N$  to  $\hat{N}$ ; there is a way to simulate a serial operation in  $N$  by a serial operation in  $\hat{N}$  (as suggested by Lemma 1 part a) provided that  $W$  is a nonnegative diagonal matrix. Look at the energy  $E_1$  of  $\hat{N}$  to be denoted by  $\hat{E}_1$ . By Lemma 1 part a:

$$E_1(t) = \hat{E}_1(2t).$$

Also,

$$E_2(t+1) = \hat{E}_1(2t+1).$$

Since  $\hat{N}$  is operating in a serial mode, it follows that  $\hat{E}_1$  is nondecreasing.

*Proof of Theorem 3 part 2:* The key in the proof is the simple observation that if a state with energy  $\hat{E}_1(t+k)$  can be reached from a state with energy  $\hat{E}_1(t)$  in  $k$  serial iterations, then it follows that  $\hat{E}_1(t) \leq \hat{E}_1(t+k)$ . If  $P_1$  and  $P_2$  in  $\hat{N}$  have the same state as  $N$  at time  $t$ , then  $\hat{E}_1(t) = E_1(t)$ . Clearly, performing one parallel iteration in  $N$  and on  $P_1$  in  $\hat{N}$  will result in  $E_2(t+1) = \hat{E}_1(t+1)$ . Hence  $E_2(t+1) \geq E_1(t)$  for every value of  $t$  when  $N$  is operating in a parallel mode. If  $N$  is in a cycle of length 2, then  $\hat{E}_1(t) = E_2(t)$ ; by using the same arguments as before it follows that  $E_2(t) \geq E_1(t)$ . If  $N$  enters a stable state at time  $t$ , then  $\hat{E}_1(t-1) = E_2(t)$  and also  $\hat{E}_1(t) = E_1(t)$ ; thus it follows that  $E_2(t) \leq E_1(t)$ .

### III. APPLICATION TO COMBINATORIAL OPTIMIZATION

Theorem 1 part 1 implies that a neural network, when operating in a serial mode, will always get to a stable state which corresponds to a local maximum in the energy function  $E_1$ . This property suggests the use of the network as a device for performing a local search algorithm for finding a local maximal value of the energy function  $E_1$  [6]. The value of  $E_1$  which corresponds to the initial state is improved by performing a sequence of random serial iterations until the network reaches a local maximum. From Theorem 1 part 2 it follows that when the network is operating in a fully parallel mode it will always reach a stable state or a cycle of length 2. The value of the energy  $E_2$  at these final points is clearly maximal with respect to the path in the state space which ends in these points. In a fully parallel operation there is no randomness in the search because there is no choice in the direction of improvement as in a serial operation.

The construction suggested by Lemma 1 enables transformation of a problem of maximization of  $E_2$  to a problem of maximization of  $E_1$ ; hence randomness can be introduced. To summarize, given a quadratic function of the form  $E_1$  or  $E_2$ , it is possible to construct a neural network which will perform a random local search for the maximum.

A rich class of optimization problems can be represented by quadratic functions [4]. A problem which not only is representable by a quadratic function but actually is equivalent to it is that of finding a minimum cut (MC) in a graph [4], [7]. In what follows, we present the equivalence between the MC problem and neural networks (Theorem 4 and 5) and also show how neural networks relate to the directed min cut (DMC) problem (Theorem 6). To make the foregoing statements clear, let us start by defining the term cut in a graph.

**Definition:** Let  $G = (V, E)$  be a weighted and undirected graph, with  $W$  being an  $n \times n$  symmetric matrix of weights of the edges of  $G$ . Let  $V_1$  be a subset of  $V$ , and let  $V_{-1} = V - V_1$ . The set of edges each of which is incident at one node in  $V_1$  and at one node in  $V_{-1}$  is called a *cut* of the graph  $G$ . A *minimum cut* in a graph is a cut for which the sum of the corresponding edge weights is minimal over all  $V_1$ .

**Theorem 4 [4], [7]:** Let  $G = (V, E)$  be a weighted and undirected graph, with  $W$  the matrix of its edge weights. Then the MC problem in  $G$  is equivalent to  $\max Q_G(X)$ , where  $X \in \{-1, 1\}^n$ , and

$$Q_G(X) \stackrel{\text{def}}{=} \sum_{i=1}^n \sum_{j=1}^n W_{ij} X_i X_j.$$

The foregoing theorem can be generalized to neural networks.

**Theorem 5:** Let  $N = (W, T)$  be a neural network with  $W$  being an  $n \times n$  zero diagonal matrix. Let  $G$  be a weighted graph with  $(n+1)$  nodes with its weight matrix  $W_G$  being

$$W_G = \begin{pmatrix} W & T \\ T^T & 0 \end{pmatrix}.$$

The problem of finding a state  $V$  in  $N$  for which  $E_1$  is a global maximum is equivalent to the MC problem in the corresponding graph  $G$ .

**Proof:** Note that the graph  $G$  is built out of  $N$  by adding one node to  $N$  and connecting it to the other  $n$  nodes with the edge connected to node  $i$  having a weight  $T_i$  (the corresponding threshold). Clearly, if the state of the added node is constrained to  $-1$ , then for all  $X \in \{-1, 1\}^n$ ,

$$Q_G(X, -1) = E_1(X).$$

Hence the equivalence follows from Theorem 4. Note that the state of node  $(n+1)$  need not be constrained to  $-1$ . There is a symmetry in the cut; that is,  $Q_G(X) = Q_G(-X)$  for all  $X \in \{-1, 1\}^{n+1}$ . Thus if a minimum cut is achieved with the state of node  $(n+1)$  being 1, then a minimum is also achieved by the cut obtained by interchanging  $V_1$  and  $V_{-1}$  (resulting in  $X_{n+1} = -1$ ).

What about directed graphs? Is it possible to design a neural network which will perform a local search for a minimum cut in a directed graph?

**Definition:** Let  $G = (V, E)$  be a weighted and directed graph. Each edge has a direction and a weight. The weights of the directed edges (arcs) can be represented by an  $n \times n$  matrix  $W$  in which  $W_{i,j}$  is the weight of the arc from  $i$  to  $j$ . Let  $V_1$  be a subset of  $V$ , and let  $V_{-1} = V - V_1$ . The set of arcs each of which has its tail at a node in  $V_1$  and its head at a node in  $V_{-1}$  is called a *directed cut* of  $G$ .

**Theorem 6 [1]:** Let  $G = (V, E)$  be a weighted directed graph with  $W$  the matrix of its edge weights ( $W$  is not necessarily

symmetric). The network  $N = (\tilde{W}, T)$  performs a local search for a DMC of  $G$  where

$$\tilde{W}_{ij} = \frac{1}{2} (W_{ij} + W_{ji})$$

$$T_k = \frac{1}{2} \sum_{i=1}^n (W_{ki} - W_{ik}).$$

The MC problem as defined in the paper is NP-hard [2]. The importance of the relation between the MC problem and neural networks lies in the fact that the MC problem can be viewed as a generic graph problem which can be mapped to the model. Thus theoretically one can transform every NP-hard problem to the MC problem and use the corresponding neural network to perform a local search algorithm. The problem with this approach is that only the problem is mapped while the algorithm for solving the problem is imposed by the way the model is operating. Theorem 6 is an example of programming the network to perform a specific local search algorithm for solving the DMC problem. It was relatively easy to find such a mapping, probably because the algorithm we chose is the one performed by the network for the MC problem.

An open problem is the following: there are many known local search algorithms for solving hard problems that have good performance; find a known local search algorithm which can be mapped to the neural network model.

## REFERENCES

- [1] J. Bruck and J. Sanz, "A study on neural networks," IBM ARC, Computer Science, RJ 5403, 1986. Also in *Int. J. Intelligent Syst.*, vol. 3, pp. 59-75, 1988.
- [2] M. R. Garey and D. S. Johnson, *Computers and Intractability, a Guide to the Theory of NP-Completeness*. San Francisco, CA: Freeman, 1979.
- [3] E. Goles, F. Fogelman, and D. Pellegrin, "Decreasing energy functions as a tool for studying threshold networks," *Disc. Appl. Math.*, vol. 12, pp. 261-277, 1985.
- [4] P. L. Hammer and S. Rudeanu, *Boolean Methods in Operations Research*. New York: Springer-Verlag, 1968.
- [5] J. J. Hopfield, "Neural networks and physical systems with emergent collective computational abilities," in *Proc. Nat. Acad. Sci. USA*, vol. 79, pp. 2554-2558, 1982.
- [6] J. J. Hopfield and D. W. Tank, "Neural computations of decisions in optimization problems," *Biol. Cybern.*, vol. 52, pp. 141-152, 1985.
- [7] J. C. Picard and H. D. Ratliff, "Minimum cuts and related problems," *Networks*, vol. 5, pp. 357-370, 1974.

## Sampling Theorems for Two-Dimensional Isotropic Random Fields

AHMED H. TEWFIK, MEMBER, IEEE, BERNARD C. LEVY, MEMBER, IEEE, AND ALAN S. WILLSKY, FELLOW, IEEE

**Abstract**—New sampling theorems are developed for isotropic random fields and their associated Fourier coefficient processes. A wavenumber-limited isotropic random field  $z(\vec{r})$  is considered whose spectral density function is zero outside a disk of radius  $B$  centered at the origin of the

Manuscript received September 3, 1986; revised February 19, 1988. This work was supported in part by the National Science Foundation under Grant ECS-83-12921 and in part by the Army Research Office under Grants DAAG-84-K-0005 and DAAL 03-86-K-0171.

A. H. Tewfik is with the Department of Electrical Engineering, University of Minnesota, 200 Union Street S.E., Minneapolis, MN 55455.

B. C. Levy is with the Department of Electrical Engineering and Computer Science, University of California, Davis, CA 95616.

A. S. Willsky is with LIDS, Massachusetts Institute of Technology, Room 35-437, Cambridge, MA 02139.

IEEE Log Number 8824062.